

9 Evolving Cooperation: From Biology to Engineering

Sabine Hauert, Sara Mitri, Laurent Keller, and Dario Floreano

9.1 Introduction

Robots are increasingly being used to solve real-world tasks such as vacuuming or assembly-line work in industrial applications. Controllers for these robots are typically designed by engineers following textbook guidelines. Although this methodology has proven to be very successful in such applications, it quickly meets its limitations as tasks become more complex. Collective robotic systems, where groups of robots cooperate to solve a distributed task in partially unknown environments, are an example of systems that are difficult to engineer following a classical approach (Beni 2004; Sahin 2005). This is because it is not obvious how to design controllers for individual robots that cooperate toward a common goal.

Evolutionary robotics (ER) has proven to be highly successful in solving difficult or underdefined engineering problems due to its potential to automatically find simple and efficient solutions (Cliff, Husbands, and Harvey 1993; Nolfi and Floreano 2000). However, for the approach to reach its full potential in solving real-world problems, we believe that a better understanding of the influence of different factors driving evolution should be developed and summarized as guidelines. An additional step is then needed to practically use the evolved controllers in a verifiable and adaptable manner.

As a starting point in constructing guidelines for the evolution of cooperative robots, we turn to the biological systems that inspired evolutionary robotics. Over billions of years, animals have evolved to solve a variety of collective tasks from navigation to collective hunting, which evolutionary biologists have studied extensively. By tapping into decades of research in biology, we explore whether the insights concerning the conditions that allow for the evolution of cooperation in nature can be translated into evolutionary algorithms that are applicable to robotic problems. For this purpose, we test the biological predictions on a preliminary robotic experiment. The results obtained from this initial study are then used as a guideline for solving a problem where we evolve a group of flying robots in simulation that must cooperate

in forming and maintaining an aerial communication network in a rescue scenario. The simple, efficient, yet unintuitive solutions discovered through this evolutionary process are then reverse-engineered and implemented in hand-designed controllers. This approach is practical for real-world applications because hand-designed controllers can be easier to understand and to parameterize for different scenarios than evolved controllers.

9.2 Understanding the Evolution of Cooperative Behavior

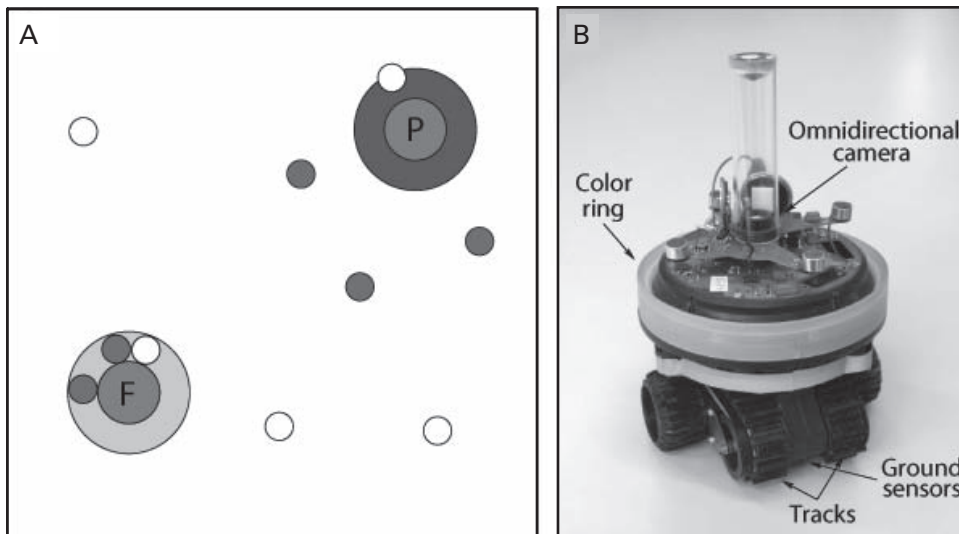
9.2.1 Cooperative Behavior in Animals

Cooperative behavior has constituted one of the biggest mysteries in evolutionary biology, and perhaps in modern biology as a whole (Dugatkin 2002; Lehmann and Keller 2006; Sachs et al. 2004; West et al. 2007). This is because the theory of Darwinian selection predicts that individuals should maximize their own reproduction, rather than altruistically increasing the reproductive success of others. However, evolutionary biology has come a long way in understanding cooperation by determining two mechanisms that may lead to the evolution of cooperative behaviors in groups of conspecifics. First, high relatedness between individuals in a group is expected to promote cooperative behavior within the group. This theory was formalized by Hamilton (1964) and is thus commonly referred to as “Hamilton’s rule,” “inclusive fitness theory,” or “kin selection theory.” A second theory that provides an explanation for the evolution of cooperative behavior is that of “group selection” (Dugatkin and Reeve 1994; Lehmann and Keller 2006; West, Griffin, and Gardner 2007). This theory states that selection *between* groups of individuals should result in cooperation *within* groups, regardless of within-group relatedness. It has recently been shown that the two theories are mathematically equivalent (Dugatkin and Reeve 1994; Hamilton 1975; Lehmann et al. 2007).

9.2.2 Cooperative Behavior in Robots

When engineering collective robotic systems, we are interested in maximizing performance. In many collective robotic tasks, performance can be increased if robots in a group cooperate toward a common goal. However, it remains unclear how groups of robots should be composed and selected to achieve maximal performance. In fact, a variety of methods are used in studies that report on evolving cooperative behavior in groups of robots (see Waibel, Keller, and Floreano 2009 for a review), yet few explicitly motivate their choice of evolutionary parameters.

To study how cooperative behavior can evolve, we designed an experimental setup consisting of groups of robots that could emit and perceive light and were evolved to solve a foraging task. Because cooperative communication can potentially increase the performance of robot groups, this system allowed us to explore whether high

**Figure 9.1**

Experimental setup. (A) A food and poison source, both emitting red light, are placed 1 m from one of two opposite corners of the square ($3\text{m} \times 3\text{m}$) arena. Robots (small circles) can distinguish the two by sensing the color of the circles of paper placed under each source using their floor sensors when driving over the paper. (B) The robot used for the experiments is equipped with two tracks to drive, an omni-directional (360 degree) vision camera, a ring of lights used to emit blue light, and floor sensors to distinguish food and poison sources.

relatedness and group-level selection can lead to an increase in cooperative behavior, as biological theory predicts, and thus an increase in group performance.

In our experimental system, ten s-bot robots (Mondada et al. 2004) placed randomly in a square arena must find a food source-emitting red light and avoid a similar red light poison source (figure 9.1). The sources could only be distinguished once the robots were very close to them, by using their floor sensors to detect a disc of colored paper placed under each of the sources. Robots could cooperate by emitting blue lights in a way that provided information on the location of the food and poison sources to other robots. Each robot was controlled using a feedforward neural network, which processed blue and red light, in addition to the information on its own location (at food, poison, or elsewhere) to determine the speeds of its two tracks and whether to emit blue light or not. The weights of the neural network formed the genome of the robot, which evolved over 500 generations in a population of 1,000 robots. The performance was calculated for each robot by counting the number of time-steps within the sixty-second trial during which it was at the food minus the number of time-steps spent by the poison, averaged over ten consecutive trials. At each generation the 200

best-performing robots were selected, replicated through cloning or crossover (with a probability of 0.2), and mutated (with a probability of 0.01 per bit) to form the next generation. For more details, see for example Floreano et al. 2007.

To test the effect of varying relatedness and the level of selection, four experimental treatments were used: (1) high relatedness, group-level selection; (2) high relatedness, individual-level selection; (3) low relatedness, group-level selection; and (4) low relatedness, individual-level selection. To form groups of low-relatedness individuals, we randomly selected 1,000 robots (with replacement) from the pool of the 200 best-performing robots and assigned them to 100 new groups of 10 robots each for the next generation. In contrast, high relatedness was achieved by selecting 100 individuals from the pool of 200, and cloning each 10 times to form the 100 new groups of 10 identical robots. Group-level selection was implemented by simply assigning the same performance score to all robots in a group that represented the average of the individual scores. Alternatively, in the individual-level selection treatments, performance was calculated independently for each individual robot (see Floreano et al. 2007 for details).

In the two treatments where relatedness between robots was high, performance was significantly higher than when relatedness was low (MannWhitney test, all $P < 0.001$, figure 9.2). Group-level selection also resulted in higher performance when robots were highly related ($P < 0.05$). However, when relatedness was low, robots selected at the group level performed significantly worse than those selected at the individual level ($P < 0.001$).

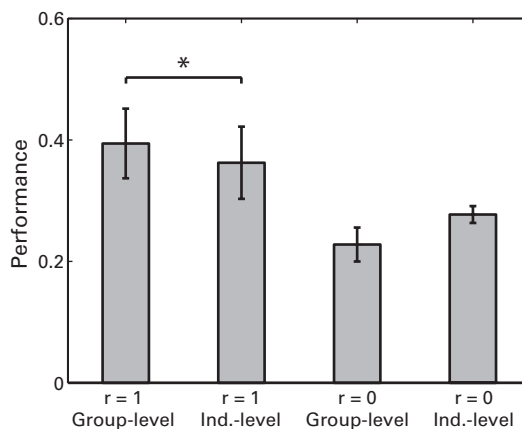
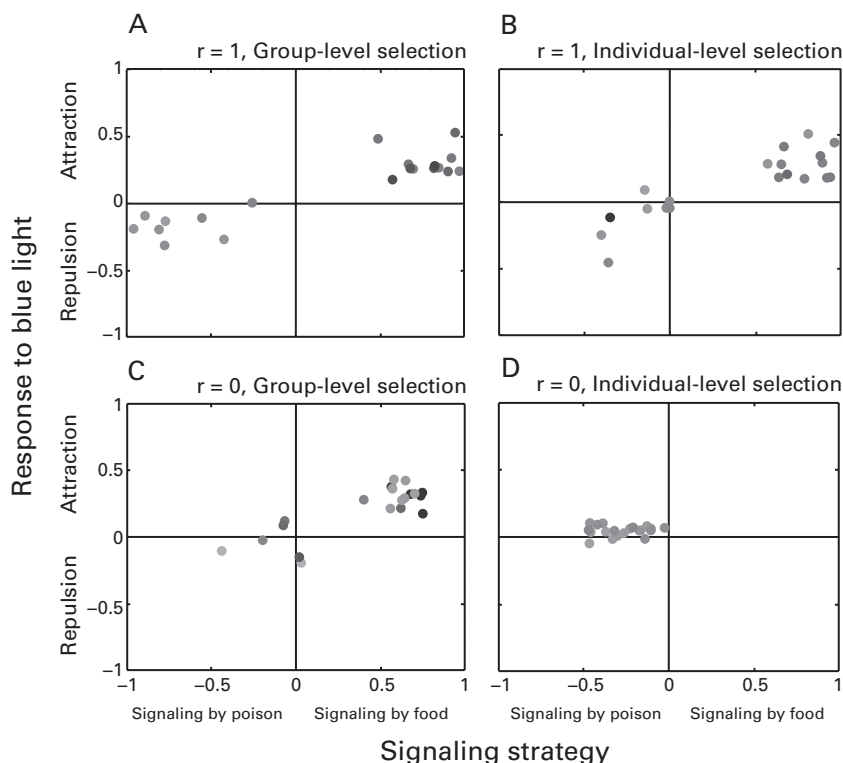


Figure 9.2

Mean (\pm SD) performance of robot populations during the last fifty generations for each of the four treatments (twenty replicates per treatment). “*” indicates that the bars are significantly different at $P < 0.05$. Bars that are not compared are significantly different at $P < 0.001$.

**Figure 9.3**

Relationship between signaling strategies and responses to blue light in the four treatments. Each dot is the average for the 100 colonies in one replicate after 500 generations of selection. Positive values for the signaling strategy indicate a tendency to signal close to the food, and negative values indicate a tendency to signal close to the poison. Positive values for the responses to blue light indicate attraction to blue light and negative values indicate repulsion (see Floreano et al. 2007 for definitions). The darkness of the points is proportional to the mean performance.

Although this performance comparison seems to indicate that unrelated robots selected at the group level did not cooperate, an analysis of their behavior shows otherwise. Both high relatedness and group-level selection led to the evolution of altruistic communication (figure 9.3a, b, and c). For these treatments, two different communication strategies were observed. In some evolutionary runs, robots produced blue light in the vicinity of the food and were attracted to blue light, thus being likely to end up by the food (e.g., top right quadrant in figure 9.3a). In other runs, blue light was more likely to be emitted by the poison, and resulted in robots driving in the opposite direction and avoiding poison (e.g., bottom left quadrant in figure 9.3a).

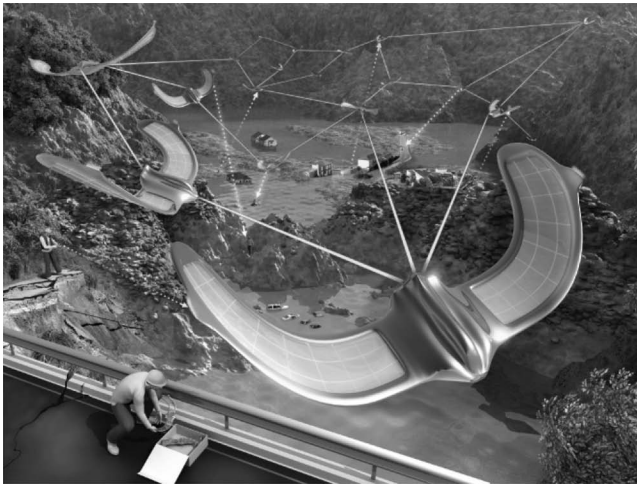
These two strategies, although not equally efficient, are both of a cooperative nature, because signalers and receivers evolved complementary strategies. However, when robots were unrelated and selected at the level of the individual, they emitted light by the poison more often than by the food, but were nevertheless slightly attracted to blue light (figure 9.3d). This strategy essentially constitutes a suppression of information, and can therefore be seen as uncooperative behavior (for more information on this strategy, see Mitri, Floreano, and Keller 2009).

These results are interesting in two respects. On the one hand, they show that the predictions of evolutionary theory hold in the case of evolutionary robotics. Since it is difficult to conduct similar experiments in living organisms, this test, within its limitations, provides some supporting contribution to theoretical discussions in biology (see Floreano et al. 2007 and Mitri, Floreano, and Keller 2009 for a discussion on this perspective). On the other hand, from an engineering perspective, it is interesting that cooperative behavior between robots does not always imply high performance and that this depends on the selection method. By designing an evolutionary algorithm using unrelated robots and selecting them at the group level, our results confirm that cooperative behavior between the robots can evolve, as expected from theoretical predictions. However, their performance has been found to be low compared to other selection methods, such as selecting related individuals at the group level. This is due to the inherent inefficiency of this particular selection algorithm. Because robots are all different in a group and the group is selected as a whole, it is difficult to select high-performing individuals, while simultaneously avoiding selecting individuals with low performance (see also “credit assignment problem” described in Waibel, Keller, and Floreano 2009). In addition, the composition of the groups changes at every generation, making it difficult for individuals to optimize their behavior with respect to the behavior of their group mates. Similar results have been obtained in a systematic study of collective object transportation by Waibel, Keller, and Floreano (2009).

In summary, we find that cooperation can evolve either if robots in a group are clones or if they share their performance scores with other members of their group. The highest performance in cooperative tasks is achieved when both these conditions are true.

9.3 From Biology to Engineering

The experiments described in the previous section allowed us to test biologically motivated theories on the evolution of cooperation, and at the same time, to draw some guidelines on the design of evolutionary algorithms for groups of cooperating robots. In this section we show how these guidelines can be applied to engineer a solution to a real-world problem. Engineering consists in finding the best possible solution to a

**Figure 9.4**

Artistic view of the use of a group of flying robots for establishing communication networks between rescuers on the ground in a flood scenario.

problem under some constraints. In the context of this chapter, this amounts to selecting the most appropriate evolutionary algorithm that is likely to lead to cooperation between robots and high performance. However, regardless of the performance of the evolved solution, it is often difficult to apply it directly to a real-world problem, because it is likely to be less predictable than a hand-engineered solution, which may result in costly failures. Alternatively, a process of reverse engineering may be applied to the evolved solution in order to derive a controller whose behavior and operating conditions are predictable while still capturing the simplicity and efficiency of the evolved solution.

To illustrate this process, we describe a situation where groups of flying robots must form communication networks between two rescuers in a disaster scenario as shown in figure 9.4. The robots are fully autonomous, such that the network of robots can be deployed by a single nonexpert rescuer on the ground. To create and maintain wireless bridges and avoid getting lost, the robots must distribute to find rescuers on the ground while staying within the communication range of one another. Flying robots have the advantage of navigating above obstacles while providing unobstructed wireless transmissions. The robots are required to work in environments with no access to GPS satellites or visual information (urban canyons, occluded environments, night operation). Therefore, they do not know their own position or the position of their neighbors. Instead they use local wireless communication and have proprioceptive sensors such as a compass and altitude and speed sensors.

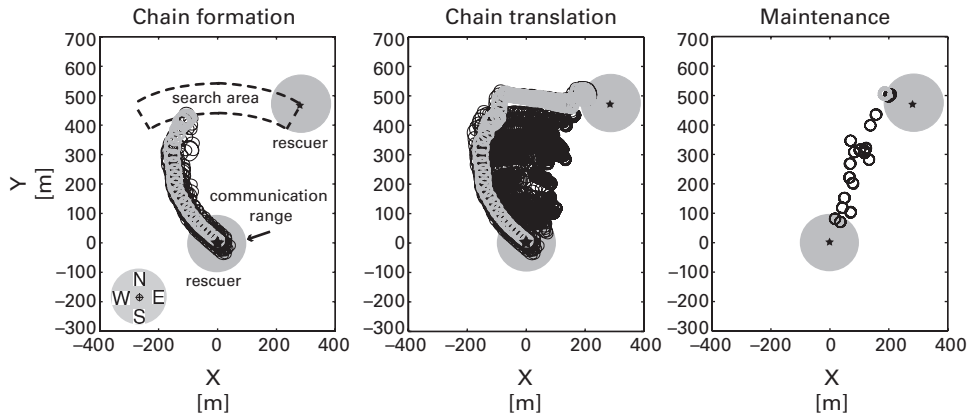
This problem is challenging because existing controllers for flying robots rely on position information and because there is no obvious strategy to design individual controllers that will lead to an effective communication network (Hauert, Zuffery, and Floreano 2009a). Furthermore, the performance of the robots can be measured only at the level of the team as a function of the quality of the resulting communication among rescuers.

9.3.1 Evolving a Group of Flying Robots

To explore this problem, we consider a simplified scenario in simulation, in which a group of robots must deploy and maintain a wireless communication network between two rescuers on the ground (Hauert, Zuffery, and Floreano 2009a). Twenty robots are launched by one rescuer at a rate of 1 every 15 s (± 7.5) and the group must then cooperate to find a second rescuer positioned within a ± 30 -degree angle of a predefined search direction and a distance of 500 ± 50 m. Once the communication link between the two rescuers is established, it must be maintained until the end of the mission, which lasts a maximum of thirty minutes. The robots are simulated using a physics engine in which we implement a first-order dynamics model of a fixed-wing robot that flies at a speed of 10 m/s and turns with a minimum turn radius of 20 m. These constraints bring interesting dynamics to the system since the robots cannot stop or turn on the spot like ground robots or hovercrafts. The communication range of robots and rescuers is of maximum 100 m with added noise between 90 m and 100 m.

Each robot is controlled using a feedforward neural controller consisting of three inputs, four hidden neurons, and one output controlling the turn rate of the robot (speed and altitude are constant). The first input to the network is the heading of the robot given by a magnetic compass. The second and third inputs are the number of network hops separating the robot from the two rescuers (high values indicate that the robot is disconnected), where network hops can be seen as the number of times a message sent from a rescuer needs to be forwarded from one robot to another before it reaches the robot in question (number of lines between a rescuer on the ground and a robot in figure 9.4). The genome of each robot consists of the 16 synaptic weights of the neural network, each represented by 8 bits, making a total genome size of 128 bits.

Based on the results obtained in the previous section, we use homogeneous groups and apply group-level selection. The performance of each group is computed as the minimum number of robots that need to fail for the communication between the rescuers to break, averaged over thirty minutes and ten missions. This performance measure favors the rapid creation of communication pathways and the robustness of the network over time. As in the foraging experiment with homogeneous individuals, a population of 100 genomes is used, which are cloned twenty times to construct 100

**Figure 9.5**

Trajectories of the robots with the best evolved controller (over all populations in all generations) during a thirty-minute mission. In this mission, the second rescuer is to the northeast of the launching rescuer to show the full extent of the chain translation displayed by the group. The trajectory of the first launched robots is shown by a light grey line.

groups of twenty robots each. After ranking the genomes according to the measured performance of the robot groups, the twenty best genomes in the population are copied to the new population (elitism) and cloned to make groups of twenty robots each. The remaining population is generated by repeatedly selecting two random individuals from the best 30 percent of the genomes, applying one-point crossover to the pair with a probability of 0.2 and then mutating the newly created individual with a probability of 0.01 per bit, and cloning it twenty times.

With these settings, artificial evolution results in controllers for flying robots that are able to create and maintain a communication network between the rescuers. At the end of the evolutionary process, robots with the best evolved controller over all populations in all generations were tested in 1,000 consecutive missions, of which 975 missions led to the creation and maintenance of a network between the two rescuers. As shown in figure 9.5, these robots form a chain that translates over the area to be searched until the second rescuer is found. The chain then stays on the spot to maintain the communication link.

9.3.2 Reverse Engineering

Robot controllers for real-world applications must often adapt across different scenarios depending on the needs of a given operation (different environment, different number of robots, slightly different task, etc.). However, evolved controllers are constrained to scenarios for which they were evolved. Furthermore, evolved controllers

such as neural networks, electronic circuits, and programs are not always easy to understand. This makes it difficult to evolve robots that are rapidly and robustly usable out of the box in unexpected situations. Possible solutions to this challenge include evolving a different controller before each operation. For this to be practical, the evolutionary process must be extremely rapid and portable. While most current evolutionary experiments are conducted over several hours or even weeks on large computer clusters, the natural increase in computational resources and power might allow for such an approach in the future. Moreover, one could imagine evolving a controller that takes as an input the parameters of the environment. This is indeed a promising approach, although it is currently challenging to find optimal controllers for the different combinations of parameters because of current limitations in evolving multi-objective systems (Urzelai and Floreano 2001). Another solution would be to allow the system to evolve online, provided that it can be given some time to fail and learn (Floreano and Mattiussi 2008). This is not necessarily obvious for all applications, including search-and-rescue missions.

Here, we propose to address this issue by reverse engineering high-performing controllers found by the evolutionary algorithm. In doing so, we aim to build a control model with a limited set of variables that captures the simplicity and efficiency of the evolved solution. To proceed, we analyze the effect of each input of the best evolved neural controller on the turn rate of the robot (see Hauert, Zuffery, and Floreano 2009b for details). Through this systematic analysis, we identify three simple behaviors performed by the individual robots:

1. Robots that are connected to the launching rescuer, even indirectly, move away from it (figure 9.6, low hop values).
2. Robots that are disconnected from the launching rescuer, move toward it with a different radius than when connected (figure 9.6, high hop values).
3. Robots connected to both rescuers turn following small circular trajectories.

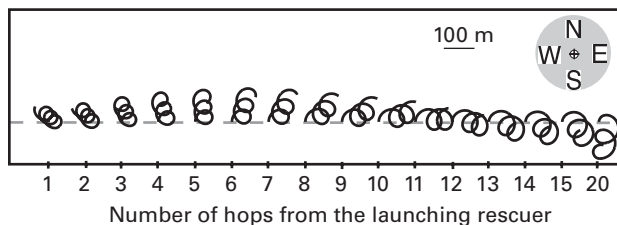


Figure 9.6

Effect of the number of hops that separate the launching rescuer from a robot on its trajectory. Here, we plot the trajectories of the best evolved controller over 30 s. Robots were never connected to the second rescuer during these experiments.

The effect of these individual behaviors on the behavior of the entire group can be hypothesized as follows. As long as robots are being launched, they remain connected to the launching rescuer (at least indirectly) while advancing in a common direction. This results in the formation of a chain. Once all robots have been launched, the chain continues to advance until it disconnects from the launching rescuer. To reconnect, the chain changes direction and moves toward the launching rescuer. Not only does the chain reconnect, but it also translates along the communication range of the rescuer, effectively sweeping through the search area and eventually finding the second rescuer. This is due to the different turn radius of the robots when disconnected or connected to the launching rescuer. Finally, robots connected to both rescuers maintain the communication pathway by performing small circular trajectories.

To explore whether the extracted behaviors yield similar collective behavior as the evolved controllers, we then translate the rules into an algorithm (Hauert, Zuffery, and Floreano 2009b). To do so, we simplify the strategy found through evolution by only considering if a robot is receiving messages from the rescuers (i.e., whether it is connected to the rescuers rather the number of hops separating it from the rescuers). The robot is assumed to fly at a constant speed v and to form circles of radius r_{\min} or more. The reverse-engineered controller, summarized as follows, outputs the turn rate ω of the robot based on the global direction ψ_s in which the robots must search for the second rescuer, the orientation of the robot ψ , and whether the robot is connected or disconnected from the rescuers.

When the robot is connected to the launching rescuer only (figure 9.7a):

$$\omega = \begin{cases} \frac{v}{r_1} & \text{if obtuse}(\psi, \psi_s) \\ \frac{v}{r_2} & \text{otherwise} \end{cases}$$

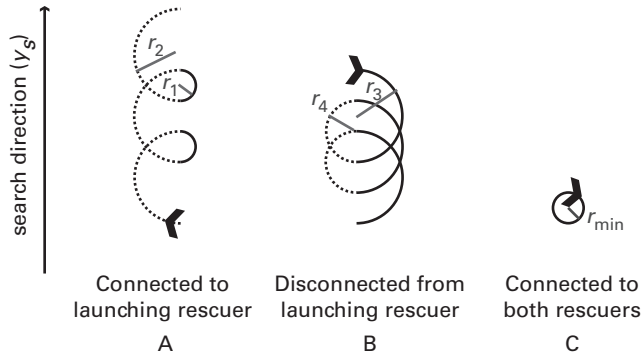
When the robot is disconnected from the launching rescuer (figure 9.7b):

$$\omega = \begin{cases} \frac{v}{r_3} & \text{if obtuse}(\psi, \psi_s + \pi) \\ \frac{v}{r_4} & \text{otherwise} \end{cases}$$

Finally, when the robot is connected to both rescuers (figure 9.7c):

$$\omega = \frac{v}{r_{\min}}$$

where obtuse returns true if there is an obtuse angle between the two variables and r_1 , r_2 , r_3 , and r_4 are parameters of the controller.

**Figure 9.7**

Robot trajectories performed by a reverse-engineered controller with parameters r_1 , r_2 , r_3 , and r_4 depending on the connection to the rescuers and the general search direction ψ_s .

Using these rules, we are able to reproduce the strategies found through evolution, namely chain formation, translation, and network maintenance. Furthermore, the rules are easier to understand than a neural network and the trajectories of the robots can be mathematically modeled and subsequently parameterized (by setting r_1 , r_2 , r_3 , and r_4) for different desired area coverage. This can be intuitively explained by the fact that the controller is based on simple geometry (circular trajectories). Therefore, changing the ratio between parameters r_1 and r_2 will lead to robot trajectories that move away from the launching rescuer at different speeds and thus affect the length of the robot chain. Furthermore, the relationship between the trajectory of a robot when connected to the launching rescuer (defined by r_1 , r_2) or disconnected (defined by r_3 , r_4) will determine the speed at which the chain translates.

9.4 Conclusion

In this chapter, we have taken inspiration from the predictions of evolutionary biology regarding the evolution of cooperative behavior, and systematically compared the effect of different evolutionary parameters on a collective system of foraging robots evolved artificially. Through these experiments, we have derived a set of guidelines, which state that cooperative behavior, as well as a high group performance can be achieved if groups are composed of genetically identical individuals and selected at the level of the group. We then applied these guidelines to a real-world problem where flying robots with limited sensory capabilities were required to create communication networks in disaster areas. Finally, we proposed to reverse engineer the best evolved solution to design a control model whose behavior is well understood and predictable.

This chapter presents the first step toward evolving groups of robots for real-world problems. Although the results shown here have been conducted in simulation, the reverse-engineered controllers already are being implemented on board real flying robots. In addition, we aim toward more complex scenarios involving windy environments, increasing the number of rescuers or allowing for mobile rescuers. Our approach has thus shown the potential of evolutionary robotics in generating efficient control solutions to complex engineering problems, such as controlling groups of robots.

More generally, we believe that the biological literature is a promising starting point to understanding many aspects of evolutionary systems. This is because many of the factors influencing systems of evolving robots have been extensively studied by biologists. The results we have reported in this chapter are only the tip of the iceberg, however. Many open questions remain on how to design evolutionary robotic systems and how to apply them to complex real-world applications. In particular, there is still a widely untapped source of biological theories that could be used for the purpose of ER, such as theories concerning division of labor, coevolution, and neuroscience. These guidelines extracted from biology to evolve controllers for robots may potentially lead to the compilation of a complete “reference manual” on how to evolve robots that can solve competitive tasks outside the lab.

Acknowledgments

This work is supported by armasuisse, competence sector Science + Technology for the Swiss Federal Department of Defense, Civil Protection and Sports; Swiss NSF grant no. K-23K0117914/1 on the “Evolution of Altruistic Communication”; and the “Swarmanoid Project,” a Future Emerging Technologies (FET IST-022888) project funded by the European Commission.

References

- Beni, Gerardo. 2004. *From Swarm Intelligence to Swarm Robotics*. *Swarm Robotics*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer.
- Cliff, Dave, Phil Husbands, and Inman Harvey. 1993. Explorations in evolutionary robotics. *Adaptive Behavior* 2:73–110.
- Dugatkin, Lee Alan. 2002. Cooperation in animals: An evolutionary overview. *Biology and Philosophy* 17:459–476.
- Dugatkin, Lee Alan, and Hudson Kern Reeve. 1994. Behavioral ecology and levels of selection: Dissolving the group selection controversy. *Advances in the Study of Behavior* 23:101–133.

Floreano, Dario, and Cladio Mattiussi. 2008. *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies*. Cambridge, MA: MIT Press.

Floreano, Dario, Sara Mitri, Stephane Magnenat, and Laurent Keller. 2007. Evolutionary conditions for the emergence of communication in robots. *Current Biology* 17:514–519.

Hamilton, William Donald. 1964. The genetical evolution of social behaviour. *Journal of Theoretical Biology* 7:1–52.

Hamilton, William Donald. 1975. Innate social aptitudes of man: An approach from evolutionary genetics. In *Biosocial Anthropology*, ed. R. Fox, 133–153. London: Malaby Press.

Hauert, Sabine, Jean-Christophe Zufferey, and Dario Floreano. 2009a. Evolved swarming without positioning information: An application in aerial communication relay. *Autonomous Robots* 26 (1): 21–32.

Hauert, Sabine, Jean-Christophe Zufferey, and Dario Floreano. 2009b. Reverse engineering of artificially evolved controllers for swarms of robots. In *CEC'09 Proceedings of the IEEE Congress on Evolutionary Computation*, ed. A. Tyrrell, P. Haddow, J. Timmis, S. Smith, 55–61. Piscataway, NJ: IEEE Press.

Lehmann, Laurent, and Laurent Keller. 2006. The evolution of cooperation and altruism: A general framework and a classification of models. *Journal of Evolutionary Biology* 19:1365–1379.

Lehmann, Laurent, Laurent Keller, Stuart West, and Denis Roze. 2007. Group selection and kin selection: Two concepts but one process. *Proceedings of the National Academy of Sciences of the United States of America* 104:6736–6739.

Mitri, Sara, Dario Floreano, and Laurent Keller. 2009. The evolution of information suppression in communicating robots with conflicting interests. *Proceedings of the National Academy of Sciences of the United States of America* 106:15786–15790.

Mondada, Francesco, Giovanni C. Pettinaro, André Guignard, Ivo W. Kwee, Dario Floreano, Jean-Louis Deneubourg, Stefano Nolfi, Luca M. Gambardella, and Marco Dorigo. 2004. Swarm-bot: A new distributed robotic concept. *Autonomous Robots* 17:193–221.

Nolfi, Stefano, and Dario Floreano. 2000. *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-organizing Machines*. Cambridge, MA: MIT Press.

Sachs, Joel L., Ulrich G. Mueller, Thomas. P. Wilcox, and James J. Bull. 2004. The evolution of cooperation. *Quarterly Review of Biology* 79:135–160.

Sahin, Erol. 2005. Swarm robotics: From sources of inspiration to domains of application. In *Swarm Robotics (WS 2004)*, LNCS 3342, ed. E. Sahin, and W. M. Spears, 10–20. Heidelberg: Springer.

Urzelai, Joseba, and Dario Floreano. 2001. Evolution of adaptive synapses: Robots with fast adaptive behavior in new environments. *Evolutionary Computation* 9 (4): 495–524.

Waibel, Markus, Laurent Keller, and Dario Floreano. 2009. Genetic team composition and level of selection in the evolution of cooperation. *IEEE Transactions on Evolutionary Computation* 13:648–660.

West, Stuart A., Stephen P. Diggle, Angus Buckling, Andy Gardner, and Ashleigh S. Griffin. 2007. The social lives of microbes. *Annual Review of Ecology Evolution and Systematics* 38:53–77.

West, Stuart A., Ashleigh S. Griffin, and Andy Gardner. 2007. Social semantics: Altruism, cooperation, mutualism, strong reciprocity and group selection. *Journal of Evolutionary Biology* 20:415–432.

PROPERTY OF MIT PRESS: FOR PROOFREADING AND INDEXING PURPOSES ONLY